REPORT DOCUMENTATION PAGE

Form Approved OMB NO. 0704-0188

1. REPORT DATE (DD-MM-YYYY)	2. REPORT TYPE	3. DATES COVERED (From - To)
08-08-2011	Conference Proceeding	-

4. TITLE AND SUBTITLE
Adaptive Virtual Support Vector Machine for the Reliability Analysis of High-Dimensional Problems

5a. CONTRACT NUMBER
W911NF-09-1-0250

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER
611102

6. AUTHORS
Hyeongjin Song, K.K. Choi, Ikjin Lee, Liang Zhao, David Lamb

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES AND ADDRESSES
University of Iowa @ Iowa City
Office of Sponsored Programs
The University of Iowa
Iowa City, IA 52242 -

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
U.S. Army Research Office
P.O. Box 12211
Research Triangle Park, NC 27709-2211

10. SPONSOR/MONITOR'S ACRONYM(S)
ARO

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
56025-NS.30

12. DISTRIBUTION AVAILIBILITY STATEMENT
Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES
The views, opinions and/or findings contained in this report are those of the author(s) and should not contrued as an official Department of the Army position, policy or decision, unless so designated by other documentation.

14. ABSTRACT
In this study, an efficient classification methodology is developed for reliability analysis while maintaining the accuracy level similar to or better than existing response surface methods. The sampling-based reliability analysis requires only the classification information – a success or a failure – but the response surface methods provide real function values as their output, which requires more computational effort. The problem is even more challenging to deal with high-dimensional problems due to the curse of dimensionality. In the newly proposed virtual support

15. SUBJECT TERMS
Surrogate Model, Support Vector Machine (SVM), Sequential Sampling, Virtual Samples, Virtual Support Vector Machine (VSVM), High-dimensional Problem

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	15. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Kyung Choi
UU	UU	UU	UU		19b. TELEPHONE NUMBER: 319-335-5684

Report Title

Adaptive Virtual Support Vector Machine for the Reliability Analysis of High-Dimensional Problems

ABSTRACT

In this study, an efficient classification methodology is developed for reliability analysis while maintaining the accuracy level similar to or better than existing response surface methods. The sampling-based reliability analysis requires only the classification information – a success or a failure – but the response surface methods provide real function values as their output, which requires more computational effort. The problem is even more challenging to deal with high-dimensional problems due to the curse of dimensionality. In the newly proposed virtual support vector machine (VSVM), virtual samples are generated near the limit state function by using linear or Kriging-based approximations. The exact function values are used for approximations of virtual samples to improve accuracy of the resulting VSVM decision function. By introducing the virtual samples, VSVM can overcome the deficiency in existing classification methods where only classified function values are used as their input. The universal Kriging method is used to obtain virtual samples to improve the accuracy of the decision function for highly nonlinear problems. A sequential sampling strategy that chooses a new sample near the true limit state function is integrated with VSVM to maximize the accuracy. Examples show the proposed adaptive VSVM yields better efficiency in terms of the modeling time and the number of required samples while maintaining similar level or better accuracy especially for high-dimensional problems.

Conference Name: ASME 2011 International Design Engineering Technical Conferences & Computers and Information in En

Conference Date: August 23, 1011

Proceedings of the ASME 2011 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference
IDETC/CIE 2011
August 29-31, 2011, Washington, D.C., USA

DETC2011-47538

Adaptive Virtual Support Vector Machine for the Reliability Analysis of High-Dimensional Problems

Hyeongjin Song, K.K. Choi*, Ikjin Lee and Liang Zhao
Department of Mechanical & Industrial Engineering
College of Engineering
The University of Iowa, Iowa City, IA 52242, USA
hyesong@engineering.uiowa.edu, kkchoi@engineering.uiowa.edu, ilee@engineering.uiowa.edu
liazhao@engineering.uiowa.edu
*Corresponding author

David Lamb
US Army RDECOM/TARDEC
Warren, MI 48397-5000, USA
david.lamb@us.army.mil

ABSTRACT

In this study, an efficient classification methodology is developed for reliability analysis while maintaining the accuracy level similar to or better than existing response surface methods. The sampling-based reliability analysis requires only the classification information – a success or a failure – but the response surface methods provide real function values as their output, which requires more computational effort. The problem is even more challenging to deal with high-dimensional problems due to the curse of dimensionality. In the newly proposed virtual support vector machine (VSVM), virtual samples are generated near the limit state function by using linear or Kriging-based approximations. The exact function values are used for approximations of virtual samples to improve accuracy of the resulting VSVM decision function. By introducing the virtual samples, VSVM can overcome the deficiency in existing classification methods where only classified function values are used as their input. The universal Kriging method is used to obtain virtual samples to improve the accuracy of the decision function for highly nonlinear problems. A sequential sampling strategy that chooses a new sample near the true limit state function is integrated with VSVM to maximize the accuracy. Examples show the proposed adaptive VSVM yields better efficiency in terms of the modeling time and the number of required samples while maintaining similar level or better accuracy especially for high-dimensional problems.

KEYWORDS

Surrogate Model, Support Vector Machine (SVM), Sequential Sampling, Virtual Samples, Virtual Support Vector Machine (VSVM), High-dimensional Problem

1. INTRODUCTION

Accurate reliability analysis is of great importance for solving engineering problems. Poor reliability analysis results can lead to unreliable or overly conservative designs. Currently, the most probable point (MPP) based methods are used to obtain reliability analysis results in many engineering problems where sensitivity information is used [1-3]. However, the sensitivity is often not available or difficult to obtain accurately in complex multi-physics or multidisciplinary simulation-based engineering design applications.

Without the sensitivity, an alternative to the MPP-based method is to directly perform the probability integration numerically by carrying out computer simulations at the Monte Carlo simulation (MCS) sampling points [4]. However, this method requires a large number of response function evaluations and can be impractical in terms of computational cost.

Therefore, surrogate-based methods are used to decrease the cost while requiring no sensitivity analysis. The main advantage of the surrogate-based method is that a limited number of function evaluations are required to construct surrogate models. Many different surrogates such as the polynomial response surface (PRS), radial basis function (RBF), multivariate adaptive regression spline (MARS), support vector regression (SVR), moving least squares (MLS) and Kriging have been developed and applied to engineering problems [5-12]. These surrogates provide approximations of otherwise expensive computer simulations. Once an accurate surrogate model is generated, the direct MCS can be applied to the surrogate model to estimate the reliability with affordable computational cost. This method is called the sampling-based reliability analysis. The sampling-based method requires the decision function to determine if a prediction at a testing point

is a success or a failure. That is, only the decision between a success and a failure is used instead of the function value. In this paper, the decision function is used to express an approximated limit state. However, surrogate-based approaches usually try to obtain accurate response function values over the given domain. Therefore, the surrogate-based methods require many samples in unnecessary regions to reach the target accuracy (i.e., Mean Squared Error or R^2), and thus they actually solve more complicated problems and become inefficient [13]. The computational burden becomes heavier in high-dimensional space due to the curse of dimensionality [14-16].

On the other hand, the support vector machine (SVM), which is a classification method, only constructs an explicit decision function [14-20]. The SVM with a sequential sampling strategy which is called the explicit design space decomposition (EDSD) is tested and applied to discontinuous problems successfully [21, 22]. Even though EDSD can be also applied to continuous problems, it often converges very slowly, and thus requires a large number of samples. One of the main reasons for the inefficiency of EDSD for continuous problems is that it only uses the classification response function values rather than the function values to construct the decision function.

In this paper, a virtual SVM (VSVM) is proposed to improve the efficiency of SVM while maintaining the good features of SVM by using the available true response function values. Unlike EDSD, VSVM is developed mainly for continuous problems. The VSVM does not depend on the availability of accurate gradient information and only constructs the decision function rather than the surrogate model over the given domain. A proposed adaptive sampling method provides new samples in the vicinity of the limit state, which makes the method even more efficient. The VSVM decision function is used to evaluate the probability of failure at a given design.

Basic concepts and important features of SVM are presented in Section 2. In Section 3, the virtual sample generation method and the adaptive sampling strategy are explained. Stopping criteria are defined to stop the updating process as the decision function converges. In Section 4, recently developed EDSD and dynamic Kriging method are compared with the proposed VSVM to demonstrate the efficiency of VSVM while maintaining the accuracy. An error measure is also defined to compare the accuracy of the result. The conclusion is followed in Section 5.

2. SUPPORT VECTOR MACHINE

An SVM is a machine learning technique used for the classification of data in pattern recognition [14-22]. It has the ability to explicitly construct a multidimensional and complex decision function that optimally separates multiple classes of data. Even though SVM is able to deal with multi-class cases, only two classes – success or failure – are used in reliability analyses, and thus only a two-class classification problem will be considered in this paper. Good features for the high-dimensional problem make SVM an appropriate method for the formulation of the explicit limit state function. In this section, a brief overview of SVM is presented, including basic ideas and some important features.

2.1 Linear SVM

For the given multidimensional problem, N samples are distributed within the local or global window. Each sample $\mathbf{x_i}$ is associated with one of two classes characterized by a value y_i=±1, which represents a success $(+1)$ or a failure (-1). The SVM algorithm constructs the decision function that optimally separates two classes of samples. The corresponding explicit boundary function is expressed as

$$s(\mathbf{x}) = b + \sum_{i=1}^{N} \alpha_i y_i K(\mathbf{x_i}, \mathbf{x}) \qquad (0)$$

where b is the bias, α_i are Lagrange multipliers obtained from the quadratic programming optimization problem used to construct SVM, $\mathbf{x}$ is an arbitrary point to be predicted, and K is the kernel of SVM. The classification of any arbitrary point $\mathbf{x}$ is given by the sign of s in Eq. (1). The optimization process is used to solve for the optimal SVM decision function with a maximal margin. Figure 1 shows a linear SVM result and the notion of margin can be easily noticed. In this case, the margin is the distance between two parallel hyperplanes given by $s(\mathbf{x})$ = ±1 in the design space. These hyperplanes are called support hyperplanes and pass through one or several samples, which are called support vectors. The SVM optimization process also does not allow any samples to exist within the margin space.

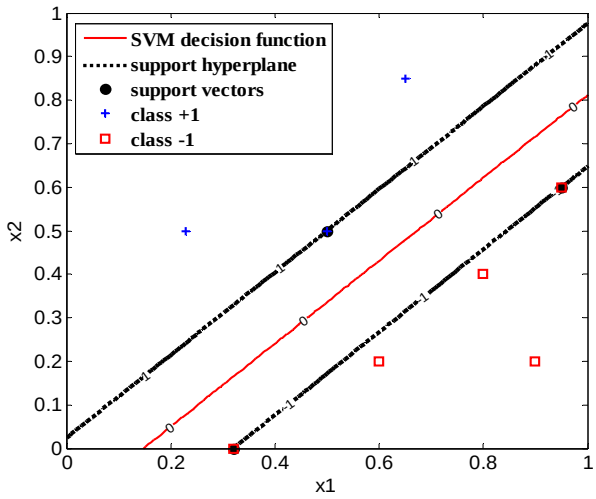


Figure 1. Linear decision function for two-dimensional problem

The Lagrange multipliers associated with the support vectors are positive while the other Lagrange multipliers equal zero. It means that the explicit SVM decision function uses only support vectors in its formulation, and thus SVM constructed only with support vectors is identical to the one obtained with all samples. Typically, the number of support vectors is much smaller than the number of samples N.

2.2 Nonlinear SVM and Kernel Functions

To construct nonlinear decision functions, kernels are introduced in SVM. In the formulation of the SVM decision function, it is assumed that there exists always a higher dimensional space where the transformed data can be linearly separated. The transformation from the original design space to the higher dimensional space is based on the kernel function K in SVM. The kernel K in SVM equation can have different

forms such as polynomial, Gaussian, Sigmoid, etc. A Gaussian kernel is used in this paper and is given as [15, 18, 19]:

$$K(\mathbf{x}, \mathbf{x}_i) = \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}_i\|^2}{2\sigma^2} \right) \quad (0)$$

where σ is the parameter of the Gaussian kernel. Figure 2 is an example of nonlinear SVM decision function with the Gaussian kernel for a two-dimensional problem. Even though the boundary is always linear in the transformed higher dimensional space, the boundary is nonlinear in the original design space. *SVM and Kernel Methods Matlab toolbox* [23] is used for the formulation of SVM.

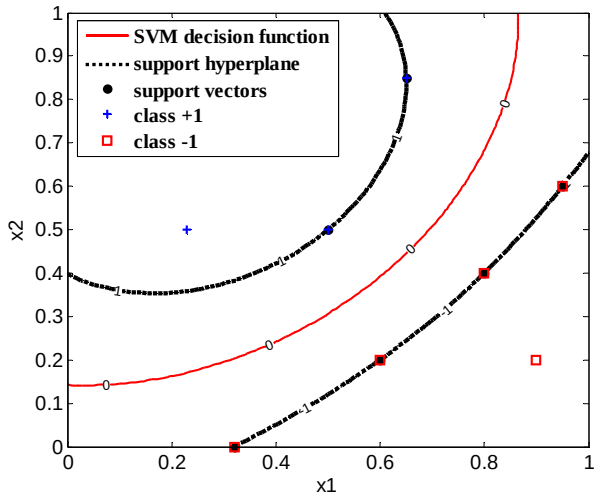


Figure 2. Nonlinear decision function for two-dimensional problem

The SVM can deal with high-dimensional problems and can separate two classes of data with the maximal margin. The SVM decision function has an explicit form, and thus predictions based on SVM are faster than those based on implicit surrogate methods such as Kriging. The prediction speed is important for sampling-based reliability analyses, since a very large number of MCS samples are required in evaluating the probability of failure.

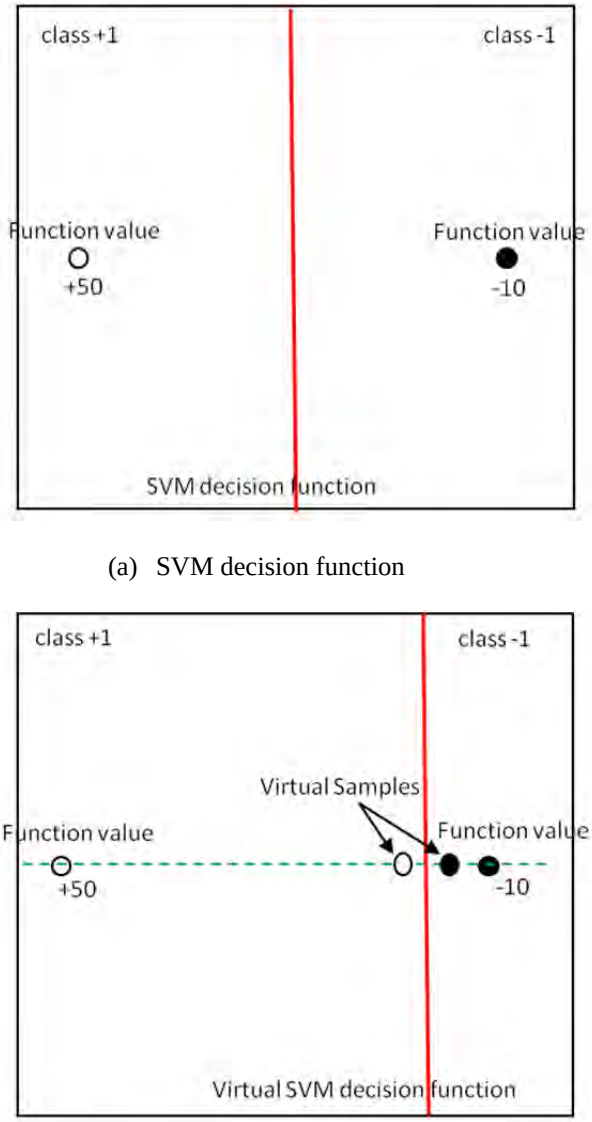
The EDSD, which is an SVM with a sequential sampling strategy, yields good performance for discontinuous limit state functions. However, EDSD is slow in convergence and requires many samples for continuous problems, since EDSD does not use function values. This can be improved by inserting virtual samples generated based on available function values.

3. VIRTUAL SUPPORT VECTOR MACHINE

3.1 Virtual Sample Generation and VSVM

For the construction of SVM, initial samples, which include both success and failure samples, should be given. Initial samples are generated by Latinized Centroidal Voronoi Tessellation (LCVT), since it shows very good uniformity and randomness [21, 27].

The classification methods such as SVM only deal with classification of responses, i.e., successes (+1) or failures (–1). The SVM decision function is located in the middle of opposite signed samples, regardless of the function values of the given samples as shown in Fig. 3 (a). However, in reality, samples with small absolute function values are more likely to be located closer to the limit state function than those with large absolute function values.



(a) SVM decision function

(b) VSVM decision function with virtual samples

Figure 3. SVM decision function and VSVM decision function with virtual samples – red solid line

The basic idea of VSVM is to increase the probability of locating the decision function close to the limit state function, by inserting two opposite signed virtual samples between the given two samples. These virtual samples play two major roles in VSVM. One is to make the predictions more accurate and the other is to locate new sequential samples near the limit state function, which will be presented in Section 3.2. In Fig. 3 (b), the VSVM decision function is shifted towards the sample with a small absolute function value by inserting two virtual samples. The virtual samples with opposite signs should be near the limit state function and be equally distanced from the limit state function to obtain the best SVM decision function.

In this paper, two types of samples are used. The first types are real samples, which include initial samples and

sequential samples. Sequential samples are inserted when the VSVM model is not accurate enough. These real samples require function evaluations. The second types are virtual samples which are generated to improve accuracy of the resulting VSVM decision function. Such virtual samples do not require function evaluations and only have virtual signs.

3.1.1 Informative Sample Set and Valid Distance

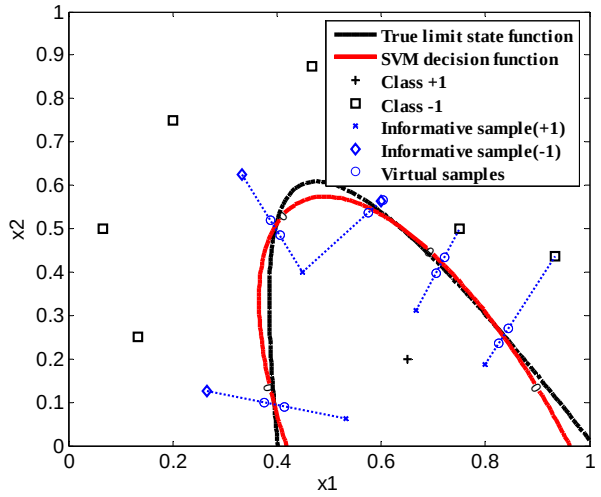
Virtual samples are generated from approximations using any pair of samples. However, it is very much desirable to use two opposite class samples. If both samples have the same sign, then finding the decision function is an extrapolation problem of which a solution is often inaccurate and is not located between two given samples. If two existing samples have opposite signs (+1 and −1), then the decision function should exist between the two samples for a continuous problem. Any pair of different class samples can be used in theory, but if the distance between two given samples is large or both samples are far from the limit state function, then the accuracy of positioning the zero point between two samples cannot be expected. Thus, one of two points should be close to the limit state function and both should be close to each other to make approximations more accurate and useful. Therefore, an informative sample set from which virtual samples are generated is defined first. Support vectors are located near the limit state function, and thus they are included in the informative set. Original SVM is constructed first based on existing samples to identify support vectors. It is highly probable that some samples with small absolute values are also located close to the limit state function, even though they may not be support vectors. All the samples that have absolute response values that are smaller than the maximum absolute responses of the support vectors are chosen as members of the informative set. This can be expressed as

$$\{\mathbf{x}_i \mid |y(\mathbf{x}_i)| \le \max_{\mathbf{x}^*}(\{|y(\mathbf{x}^*)|\}), i = 1, \cdots, N\} \qquad (0)$$

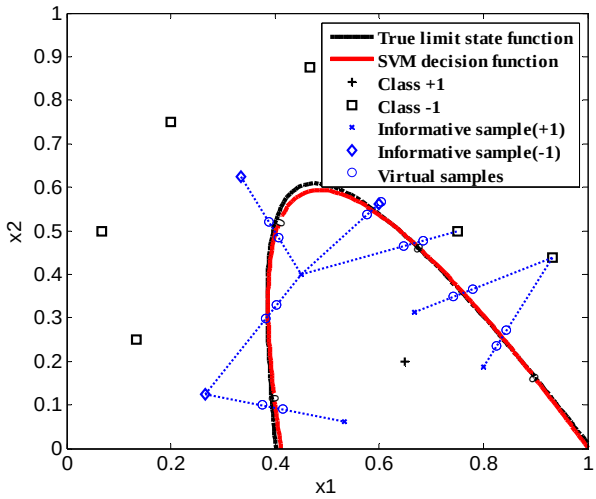
where $\mathbf{x}_i$ is the i^{th} sample, $\mathbf{x}^*$ are support vectors, y is the function value at the given position, N is the number of samples and $\{|y(\mathbf{x}^*)|\}$ is a set of absolute response function values at support vectors.

From the previously chosen informative samples, the closest opposite signed samples are paired to generate virtual samples between each pair. However, there exist some pairs that can generate important virtual samples, even though they are not the closest opposite signed samples to each other. To solve this problem, a valid distance concept is introduced. Pairs can generate virtual samples if the distance between them is shorter than the valid distance. If the valid distance is too large, then there is a risk of including many unnecessary virtual samples and producing poor approximations. If the valid distance is too short, it may not include more useful information. Figure 4 shows the influence of the valid distance concept in a two-dimensional example. By inserting an additional pair of virtual samples between two existing virtual sample pairs, the accuracy is improved in the area near the new virtual sample pair.

The distances between pairs of informative samples and the closest opposite signed samples can be obtained. The maximum distance between above pairs is defined as the valid distance in this paper.



(a) The closest samples only (without the valid distance concept)



(b) With the valid distance concept

Figure 4. VSVM decision functions with/without the valid distance concept

3.1.2 Approximations for Zero Positions

Two additional steps are needed for the generation of the virtual samples after the informative sample set and the valid distance are defined. Firstly, since the true limit state function is not known in general, a zero position is approximated from two different class samples by using approximation methods such as linear approximation, Kriging or MLS. A zero position means a point where the approximation value is zero among all the points on the line between two opposite signed samples. Linear approximation simply assumes that the function value between two given samples is linear and tries to find the zero point. Linear approximation is very fast and easy to apply but can be inaccurate for highly nonlinear functions.

Since new samples are located near the true limit state function by the sequential sampling method, the Kriging or MLS methods, which are accurate near given samples, are appropriate to obtain better approximations. In this paper, the universal Kriging method is used to approximate the zero

point between two opposite signed samples and SURROGATES toolbox [24] is used for the construction of the universal Kriging model. The optimization problem for finding the zero position between two samples is expressed as

$$\min_{\mathbf{x}} \left| \hat{A}(\mathbf{x}) \right|$$
$$s.t. \quad \mathbf{x} = \mathbf{x}_\mathbf{i} \cdot t + \mathbf{x}_\mathbf{j} \cdot (1-t) \qquad (0)$$
$$0 \le t \le 1$$

where $\mathbf{x}_i$ and $\mathbf{x}_j$ are original samples with opposite signs, $\mathbf{x}$ is a point on the straight line connecting $\mathbf{x_i}$ and $\mathbf{x_j}$ and $\hat{A}(\mathbf{x})$ is an approximated value at $\mathbf{x}$ obtained by the universal Kriging method.

When new sequential sample is inserted, the universal Kriging model is constructed based on the new sample set. In the Kriging model, the correlation function $R(\boldsymbol{\theta}, \mathbf{x}_i, \mathbf{x}_j)$ should be estimated from the sample data, where $\mathbf{x}_i$ and $\mathbf{x}_j$ are given samples and $\boldsymbol{\theta}$ is the process parameter. The influence of the parameter $\boldsymbol{\theta}$ on the performance is significant, and thus the determination of the parameter is important. To find the optimum $\boldsymbol{\theta}$, different methods such as Hookes&Jeeves (H-J), Lavenberg-Marquardt (L-M), genetic algorithm (GA) and PatternSearch (PS) methods [24, 25] have been applied. Among them, the PS method is most accurate but it requires more computational effort than other methods. However, with VSVM, less number of iterations can be used to achieve a similar level of accuracy with more accurate Kriging models by locating new samples correctly. Therefore, time and resources can be saved by using the PS method.

To make the estimation process more efficient, the history of parameter changes was investigated to find that new optimum $\boldsymbol{\theta}$ is close to the previous optimum $\boldsymbol{\theta}$ with one less sample in general. If the current SVM model is similar to the previous SVM then both optimum Kriging parameters are also close to each other. Therefore, the previous optimum Kriging parameter $\boldsymbol{\theta}$ value is used as the initial value for the PS method. By implementing this efficiency strategy, the elapsed time to find the optimum $\boldsymbol{\theta}$ is reduced by 90% per iteration in average.

It requires fair amount of computational time to solve Eq. (4) accurately. However, if the zero position is within the virtual margin explained in Section 3.1.3, then the resulting SVM decision function is similar to the decision function with exact zero position. Also Kriging approximations take large amount of time if approximations are calculated one by one due to its implicit formulation. Therefore, the line connecting two opposite signed samples $\mathbf{x}_i$ and $\mathbf{x}_j$ is divided into 100 elements, their Kriging approximations are evaluated at once and the position with the minimum absolute function value is chosen. 100 elements are used in this paper because the virtual margin is 0.02 and the mean distance between existing sample pairs is 1.2 in the normalized variable space. By introducing the new method, the elapsed time for generating virtual samples is reduced from 39.94 sec. to 2.01 sec. per iteration for the twelve-dimensional problem.

3.1.3 Generation of Virtual Samples from Zero Positions

Secondly, two opposite signed virtual samples are generated near the zero point. One is located in the direction of the success sample and the other is in the direction of the failure sample. These are virtual samples and the one shifted towards the success sample will be assigned as a success and the other one will be assigned as a failure virtually. Both virtual samples should be between the given two opposite signed samples and on the line that connects these points. Then, a new SVM decision function based on the original and virtual samples will be located between the virtual sample pairs, because the virtual samples in the pair have different signs and are close to each other. If approximations for zero points are accurate, then both virtual samples and a new decision function will be near the limit state function.

One important question is how closely a pair of virtual samples should be located. If the distance between a pair of virtual samples is too long, then these virtual samples will not be chosen as support vectors and they become meaningless. To make the virtual samples useful, the distance should be short enough so that the virtual samples are chosen as support vectors. However, due to the error of the sampling-based probability of failure evaluation [1], the virtual margin, the distance between a pair of virtual samples should not be extremely small. Therefore, a decision about the size of the virtual margin should be based on the target error level.

If many virtual samples are clustered together within a small region, the additional information from most closely located virtual samples is negligible and the computational time increases unnecessarily. In each virtual sample choice process, both the amount of additional information and the computational cost should be considered. The first pair of virtual samples are generated between a sample with the smallest absolute function value and its closest opposite signed sample, since they provide the most accurate approximations.

After the first pair is chosen, the valid distance is defined based on SVM with initial sample set, and virtual sample candidates are generated from two opposite samples within the valid distance. The candidate pair that have the longest distance from both real and virtual samples are chosen as the next virtual samples to prevent clustered virtual samples within a small region. To avoid clustered virtual samples, the number of virtual samples is limited by a predefined number. Otherwise, the process will end up generating unnecessarily many virtual samples.

Once all virtual samples are generated, new VSVM can be constructed by using both existing samples and virtual samples.

3.2 Adaptive Strategy with Sampling and Stopping Criteria

3.2.1 Adaptive Sequential Sampling

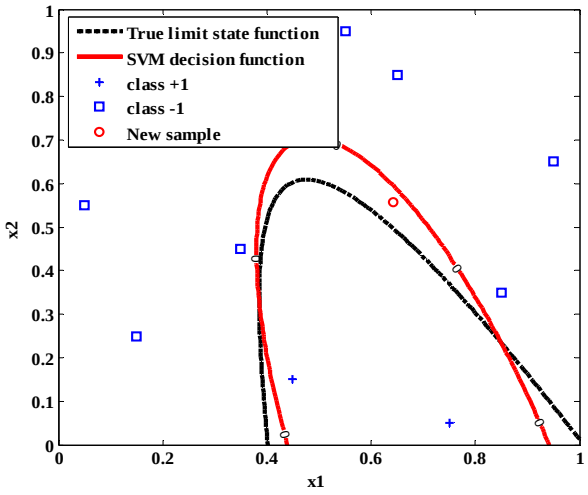
The surrogate-based approaches construct a model that is accurate over the given domain, and thus samples tend to spread out within the given domain to satisfy the target accuracy. However, since only an accurate decision function is required for the sampling-based methods, samples near the limit state function are more informative than samples far away from the limit state function. Such efficiency cannot be achieved by using a uniform sampling strategy, and thus a sequential sampling method is crucial for better efficiency and accuracy.

In this paper, a new sample is selected such that it is located within the margin ($|s(\mathbf{x})|<1$), which is narrow since each pair of virtual samples are closely located. In addition, a

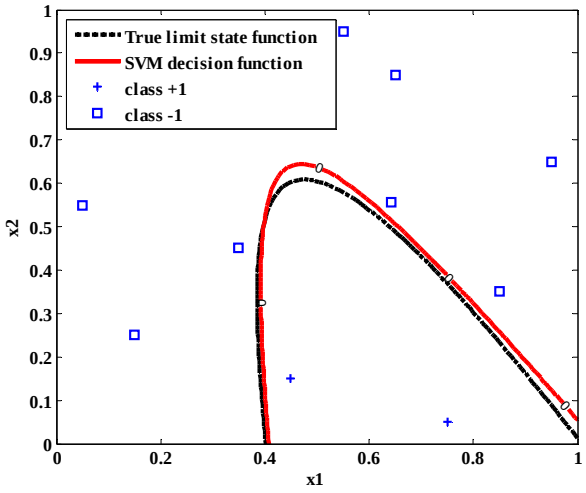
new sample should have the maximum distance from the closest existing sample to maximize the additional information by the new sample. This strategy is similar to the sequential sampling method by Basudhar and Missoum [21] but the computational burden can be reduced by using the within-the-margin constraint ($|s(\mathbf{x})|<1$) rather than the on-the-decision-function constraint ($s(\mathbf{x})=0$) which is more difficult to satisfy. A less strict constraint can be used with VSVM since new samples do not need to be on the limit state function by introducing virtual samples. In other words, if new samples are located near the limit state function, accurate virtual samples close to the limit state function can be obtained. The optimization problem is defined as

$$\max_{\mathbf{x}} \left\| \mathbf{x} - \mathbf{x}_{nearest} \right\| \quad s.t. \left| s(\mathbf{x}) \right| < 1 \tag{0}$$

where $\mathbf{x}_{nearest}$ is the existing sample closest to the new sample $\mathbf{x}$. Since $\mathbf{x}_{nearest}$ changes as the position of new sample candidate $\mathbf{x}$ moves, Eq. (5) is a moving target problem. In Fig. 5, new sample is inserted into a region near the limit state function and where there is no existing sample nearby. The VSVM decision function is improved drastically near the sequential sample.



(a) The VSVM decision function and a sequential sample



(b) The VSVM decision function with a new sample

Figure 5. Changes of the VSVM decision function in the normalized design space

As explained in the previous paragraph, the accurate solution for Eq. (5) is not necessary. Therefore, gradient-based optimization methods such as trust-region-reflective algorithm [28, 29], active-set algorithm [30, 31] or interior-point algorithm [32, 33] can be used instead of the PS method since they are faster than PS without sacrificing the accuracy much.

3.2.2 Stopping Criteria

Stopping criteria are required to determine when the decision function is converged. Since the true limit state function is not known, the criterion is based on the variations of the approximated decision function. A set of N_{stop} testing points is generated using input distributions because the MCS samples are also generated in the same way for the sampling-based reliability analysis. In this paper, ten thousand testing samples were used for all examples. The fraction of testing points that show different signs from the previous surrogate is calculated as [21]

$$\Delta_k = \frac{\sum_{i=1}^{N_{stop}} I_k(\mathbf{x}_i)}{N_{stop}} \times 100(\%) \tag{0}$$

where k is the current iteration number, Δ_k is the fraction of testing points for which the sign of the SVM evaluation changes between k-1th and k^{th} iterations. $I_k(\mathbf{x}_i)$ in Eq. (6) is an indicator function defined as

$$I_k(\mathbf{x}_i) \equiv \begin{cases} 1, \left| sign(s_{k-1}(\mathbf{x}_i)) - sign(s_k(\mathbf{x}_i)) \right| > 0 \\ 0, \text{otherwise} \end{cases} \tag{0}$$

where $s_{k-1}(\mathbf{x}_i)$ and $s_k(\mathbf{x}_i)$ represent the SVM value at $\mathbf{x}_i$ at k-1th and k^{th} iterations, respectively. Changes in the SVM decision function fluctuate and usually decrease as the number of iterations increases as is shown in Fig. 6.

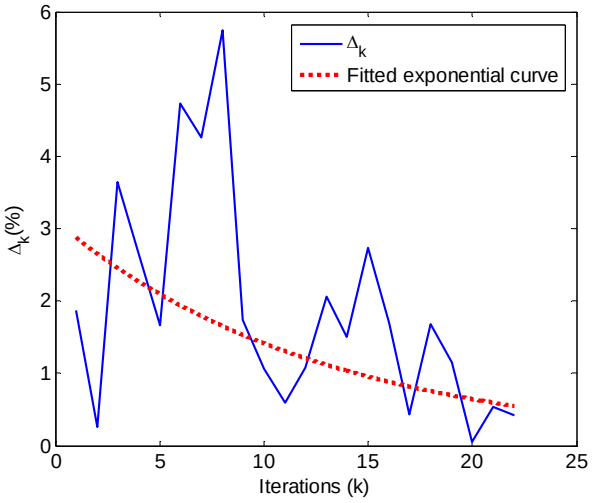


Figure 6. Changes of Δ_k and fitted exponential curve

In order to implement more stable stopping criteria, the fraction of testing points changing signs between successive iterations is fitted by an exponential curve as [21]

$$\hat{\Delta}_k = Ae^{Bk} \tag{0}$$

where $\hat{\Delta}_k$ represents the fitted values of Δ_k and A and B are the parameters of the exponential curve. The value of $\hat{\Delta}_k$ and the slope of the curve are calculated whenever each new sample is added. If Δ_k is large while $\hat{\Delta}_k$ is small, it means that a big change occurred in the model at the k^{th} iteration, which $\hat{\Delta}_k$ did not catch. If Δ_k is small while $\hat{\Delta}_k$ is large, the situation is that the new sample is inserted into a region where zero-position approximations are already accurate, so there is a small change between recent two models but it may not be converged yet. Therefore, both Δ_k and $\hat{\Delta}_k$ should be kept small for more robust results. The slope of the curve is also kept close to zero for stable results.

To stop the updating process, the maximum of Δ_k and $\hat{\Delta}_k$ should be less than a small positive number ε_1. Simultaneously, the absolute value of the slope of the curve at convergence should be lower than ε_2. Thus, the stopping criteria can be defined as

$$\begin{aligned} &\max(\Delta_k, \hat{\Delta}_k) < \varepsilon_1 \\ &-\varepsilon_2 < BAe^{Bk} < 0. \end{aligned} \tag{0}$$

ε_1 and ε_2 are determined so that the target classification error level can be achieved. The target classification error is 2.0% in this paper. For more accurate limit state function, smaller values can be applied. Generally, ε_2 should be smaller than ε_1 for more stable convergence.

The overall procedure of VSVM with a sequential sampling strategy is shown as Fig. 7.

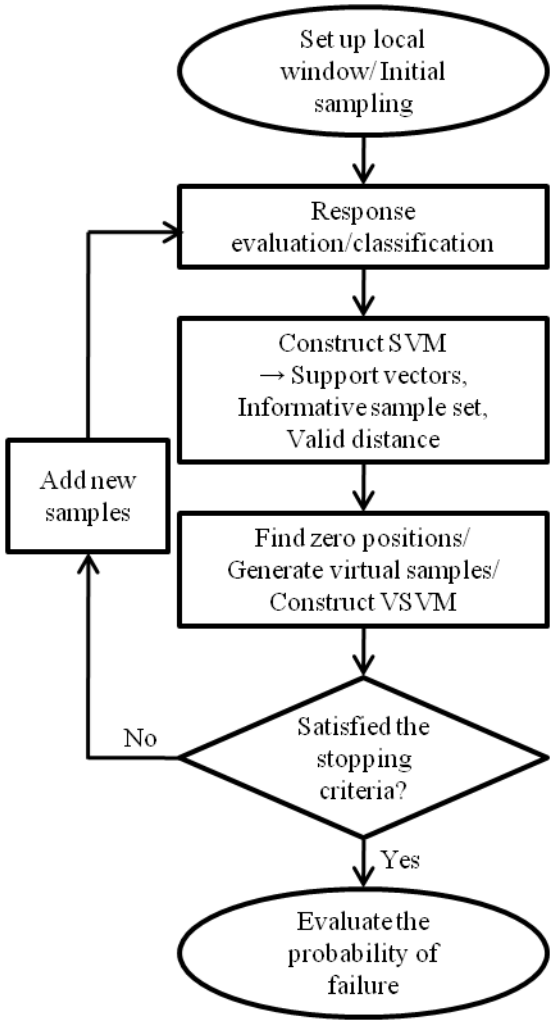


Figure 7. Flowchart of VSVM with a sequential sampling strategy

4. COMPARISON STUDY BETWEEN VSVM AND OTHER SURROGATES

4.1 Comparison Procedure

The two most recent surrogate modeling methods with sequential sampling schemes were selected to be compared with the proposed VSVM. One is the explicit design space decomposition (EDSD) method with an improved adaptive sampling scheme that uses SVM [21, 22]. The improved adaptive sampling method has two ways to choose a new sample: (1) to select the sample that has the largest distance to the closest existing samples while maintaining $s(\mathbf{x})=0$, and (2) to choose the support vector $\mathbf{x}^*$ that is farthest from the existing samples of the opposite class and to select the sample that is farthest from $\mathbf{x}^*$ while maintaining the opposite sign of y^* and on the hypersphere of radius R centered around $\mathbf{x}^*$. y^* is the function value at $\mathbf{x}^*$ and R is chosen as half the distance from $\mathbf{x}^*$ to the closest opposite signed sample. For a fair comparison for both EDSD and VSVM, the same parameters for SVM are used. Therefore, the differences between them are the sequential sampling strategy and the use of virtual samples.

The other surrogate modeling method is the dynamic Kriging (DKG) with a sequential sampling method [12]. Zhao

et.al., showed that DKG is one of the most accurate response surface methods when the same number of samples is used. DKG was compared with polynomial response surface, radial basis function, support vector regression, and universal Kriging. Therefore, dynamic Kriging is chosen to compare the accuracy of VSVM with one of best response surface methods. The basic form of the dynamic Kriging prediction is expressed as

$$\hat{y}(\mathbf{x}) = (\mathbf{r}_0 - \mathbf{F}\boldsymbol{\lambda})^T \mathbf{R}^{-1}\mathbf{Y} \quad (0)$$

where $\mathbf{R}$ is the symmetric correlation matrix, $\mathbf{r}_0$ is the correlation vector between the prediction location $\mathbf{x}$ and all N samples $\mathbf{x}_i$, i=1,⋯,N, $\mathbf{Y}$ is the response vector, $\mathbf{F}$ is a design matrix of basis functions and $\boldsymbol{\lambda}$ is a regression coefficient vector. In the dynamic Kriging method, $\mathbf{F}$ is not fixed, but the best one is chosen by the genetic algorithm (GA). The sequential sampling method chooses a new sample where the prediction variance is largest.

Three test examples are used to show the performance of the adaptive sampling-based VSVM. One example is a low-dimensional problem and the other two are high-dimensional problems. SVM can be applied to both global and local windows. However, global window usually requires unnecessarily many samples to achieve the target accuracy in reliability analyses. Therefore, SVM is applied to local windows of the original input domain and the original functions are shifted appropriately to include both signed samples so that the local windows include the true limit state functions. In Section 4.2, 4.3 and 4.4, local windows are defined as hyper-cubes based on lower and upper bounds respectively.

For the Gaussian kernel in Eq. (2), parameter σ should be provided. Decision of optimum σ is an ongoing research subject. In this paper, fixed σ values, which are small enough to maintain zero training error, are used. Training error is defined as the classification error with respect to existing samples and not testing samples.

Since the SVM is a classification method and only takes care of the decision function, the mean squared error (MSE) and R^2, which are widely used in the surrogate-based methods, cannot be used for comparison. Therefore, the accuracy of the SVM decision function should be judged by its closeness to the true limit state function. In real situations, the limit state function is often unavailable and so is the error measure. However, the error measure can be obtained for academic analytical test functions. One million testing points (N_{test}) are generated based on input distributions because the MCS samples are also generated in the same way for the sampling-based reliability analysis. These testing points are used to calculate the classification error, which is the fraction of misclassified testing points over total number of testing points. A test point for which the sign of SVM does not match the sign provided by the true limit state function is considered as misclassification [21]. Therefore, the classification error c is

$$c = \frac{\sum_{i=1}^{N_{test}} I(\mathbf{x}_{test})}{N_{test}} \times 100(\%) \quad (0)$$

where $\mathbf{x}_{test}$ represents a test sample. $I(\mathbf{x}_{test})$ in Eq. (11) is an indicator function and defined as

$$I(\mathbf{x}_{test}) \equiv \begin{cases} 1, & s(\mathbf{x}_{test}) \cdot y_{test} < 0 \\ 0, & otherwise \end{cases} \quad (0)$$

where y_{test} represents the corresponding classification value (±1) at $\mathbf{x}_{test}$, $s(\mathbf{x}_{test})$ is the SVM approximation at $\mathbf{x}_{test}$.

Our purpose is to evaluate the probability of failure accurately. The relationship between the probability of failure measurement error and the classification error is approximately proportional. Therefore, accurate probability of failure can be obtained by keeping the classification error small. Also, the classification error represents the accuracy of the obtained limit state function, so the classification error is used as the error measure for comparison in this paper.

4.2 2-D Example

The analytic function is a 4th order polynomial function, which is expressed as

$$\begin{aligned} f(\mathbf{x}) = 1 + (0.9063 \cdot x_1 + 0.4226 \cdot x_2)^2 + (0.9063 \cdot x_1 \\ +0.4226 \cdot x_2 - 6)^3 - 0.6(0.9063 \cdot x_1 + 0.4226 \cdot x_2)^4 \\ -(-0.4226 \cdot x_1 + 0.9063 \cdot x_2) \\ 4.5 \le x_1 \le 6.5,\ 5.5 \le x_2 \le 7.5 \end{aligned} \quad (0)$$

The number of initial samples is 10 for all 20 tests and each test starts with different initial profiles. Parameters σ, ε_1, and ε_2 are 3, 0.8, and 0.3, respectively, for both EDSD and VSVM. To compare the performances with respect to the same number of additional samples, VSVM is performed first and DKG and EDSD are performed later using the same number of samples as VSVM. Each process is forced to stop when it reaches the same number of samples. Each method has its own sequential sampling strategy, and thus all final profiles are different except the 10 initial samples. According to Table 1, which provides averaged values of 20 test cases, EDSD is the fastest, but the classification error is not accurate at all. This clearly shows that EDSD converges slowly due to incapability of using exact response function values. The VSVM uses about the same amount of time as DKG and results in a better classification error.

Table 1. Average classification error and elapsed time over 20 tests

	DKG	EDSD	VSVM
Classification error (%)	2.5739	15.3364	0.3428
Elapsed time (sec)	35.3	3.2	33.1

4.3 9-D Example

The nine-dimensional extended Rosenbrock function is used for the test, which is expressed as

$$\begin{aligned} f(\mathbf{x}) = \sum_{i=1}^{8} \left[(1 - x_i)^2 + 100(x_{i+1} - x_i^2)^2 \right] - 68000 \\ -3 \le x_i \le -2, i = 1, \ldots, 9. \end{aligned} \quad (0)$$

The initial sample size is 20, and 20 different initial sample profiles are used. For both EDSD and VSVM, σ, ε_1, and ε_2 are 5, 0.5, and 0.03, respectively. The same number of additional samples is used in the same way as previous two-dimensional problem. In Table 2, which provides averaged values of 20 test cases, EDSD is still the fastest, but the classification error is not accurate. VSVM uses about half amount of time as DKG and results in better classification error. Therefore, VSVM is efficient and accurate for nine-dimensional problem.

Table 2. Average classification error and elapsed time over 20 tests

	DKG	EDSD	VSVM
Classification error (%)	2.3096	6.7944	1.7816
Elapsed time (sec)	196	60	103

4.4 12-D Example

For a twelve-dimensional example, the Dixon-Price function, which is expressed as

$$f(\mathbf{x}) = (x_1 - 1)^2 + \sum_{i=2}^{12} i(2x_i^2 - x_{i-1})^2 - 36000 \qquad (0)$$
$$3 \le x_i \le 4, i = 1,\dots ,12.$$

is used. The initial sample size is 35 for 20 tests. Parameters σ, ε_1, and ε_2 are 15, 0.25, and 0.015, respectively. The same number of additional samples is used for all three methods. In Table 3, which provides averaged values of 20 test cases, EDSD is the fastest, but the classification error is not accurate. VSVM uses less time than DKG but results in a better classification error.

Table 3. Average classification errors and elapsed time over 20 tests

	DKG	EDSD	VSVM
Classification error (%)	2.0176	8.8797	1.6722
Elapsed time (sec)	289	64	169

For other way of comparison, EDSD is performed using the same stopping criteria as VSVM so that EDSD can use more samples to construct the decision function. According to Table 4, the average number of additional samples of EDSD is 77.9, which is far more than 33.3 of VSVM. EDSD also uses slightly less time than VSVM, and the classification error is still quite large. Clearly, VSVM is more accurate and efficient than EDSD.

Table 4. Average number of additional samples, classification error, and elapsed time with the same stopping criteria over 20 tests

	EDSD	VSVM
Number of additional samples	77.9	33.3
Classification error (%)	6.9029	1.6722
Elapsed time (sec)	149	169

Since DKG and VSVM use different stopping criteria, a smaller stopping criterion is used for DKG to achieve a classification error similar to that of VSVM. In Table 5, DKG can achieve a classification error level similar to that of VSVM after it uses about 6 more samples. Furthermore, the elapsed time of DKG is larger than that of VSVM.

Table 5. Average number of additional samples, classification error, and elapsed time of DKG and VSVM when similar classification error was achieved (20 tests)

	DKG	VSVM
Number of additional samples	39.4	33.3
Classification error (%)	1.7381	1.6722
Elapsed time (sec)	341	169

VSVM is more efficient than DKG in terms of elapsed time for modeling while maintaining better accuracy level, especially in high-dimensional space. EDSD converges very slowly and is inefficient in terms of the number of additional samples. This is more problematic when the computer simulations at each sample point are very expensive.

For future, efficiency strategies can be modified further to make VSVM faster while maintaining the accuracy. This adaptive VSVM also will be applied to sampling-based reliability-based design optimization (RBDO).

5. CONCLUSION

A sequential sampling-based virtual support vector machine method is proposed to efficiently construct the accurate decision function for the reliability analysis, especially in high-dimensional space. Virtual samples are generated from real samples and their response function values to improve the accuracy of the SVM decision function, and the sequential sampling method is also used to increase the efficiency of the algorithm by inserting new samples near the true limit state function.

The proposed method is compared with different surrogate modeling methods such as EDSD and DKG with their own sequential sampling strategies. DKG can construct accurate surrogates with relatively small number of samples but it is inefficient since the dynamic basis selection process requires significant computational effort [12]. For a low-dimensional problem, both VSVM and DKG are accurate and require similar modeling time. However, VSVM becomes more efficient than DKG and EDSD while maintaining the required accuracy for high-dimensional problems. Therefore, both VSVM and DKG are recommended to be applied to low-dimensional problems, and adaptive VSVM is recommended for high-dimensional problems. EDSD requires a large number of samples in all cases, since it does not use function values.

6. ACKNOWLEDGEMENT

Research is jointly supported by the ARO Project W911NF-09-1-0250 and the Automotive Research Center,

which is sponsored by the U.S. Army TARDEC. These supports are greatly appreciated.

7. REFERENCES

[1] Haldar, A., and Mahadevan, S., "*Probability, Reliability and Statistical Methods in Engineering Design*," John Wiley & Sons, New York, 2000.

[2] Tu, J., Choi, K.K., and Park, Y.H., "A New Study on Reliability-Based Design Optimization," *Journal of Mechanical Design*, Vol.121, No.4, pp.557-564, 1999.

[3] Youn, B.D., Choi, K.K., and Du, L., "Enriched Performance Measure Approach for Reliability-Based Design Optimization," *AIAA Journal*, Vol.43, No.4, pp.874-884, 2005.

[4] Rubinstein, R.Y., "*Simulation and the Monte Carlo method*," Wiley, New York, 1981.

[5] Cressie, N.A.C., "*Statistics for Spatial Data*," John Wiley & Sons, New York, 1991.

[6] Barton, R.R., "Metamodeling: a State of the Art Review," *WSC '94: Proceedings of the 26th Conference on Winter Simulation, Anonymous Society for Computer Simulation International*, San Diego, CA, USA, pp.237-244, 1994.

[7] Jin, R., Chen, W., and Simpson, T., "Comparative Studies of Metamodelling Techniques Under Multiple Modelling Criteria," *Structural and Multidisciplinary Optimization*, Vol.23, No.1, pp.1-13, 2001.

[8] Simpson, T., Poplinski, J., and Koch, P., "Metamodels for Computer-Based Engineering Design: Survey and Recommendations," *Engineering with Computers*, Vol.17, No.2, pp.129-150, 2001.

[9] Wang, G.G., and Shan, S., "Review of Metamodeling Techniques in Support of Engineering Design Optimization," *Journal of Mechanical Design*, Vol.129, No.4, pp.11, 2007.

[10] Forrester, A., Sobester, A., and Keane, A., "*Engineering Design via Surrogate Modelling, A Practical Guide*," John Wiley & Sons, United Kingdom, 2008.

[11] Forrester, A., and Keane, A., "Recent Advances in Surrogate-Based Optimization," *Progress in Aerospace Sciences*, Vol.45, No.1-3, pp.50-79, 2009.

[12] Zhao, L., Choi, K.K., and Lee, I., "A Metamodel Method Using Dynamic Kriging and Sequential Sampling," *The 13th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Fort Worth, TX, Sept.13-15, 2010.

[13] Hurtado, J.E., and Alvarez, D.A., "Classification Approach for Reliability Analysis with Stochastic Finite-Element Modeling," *Journal of Structural Engineering*, Vol.129, No.8, pp.1141-1149, 2003.

[14] Vapnik, V.N., "*Statistical Learning Theory*," Wiley, New York, 1998.

[15] Cherkassky, V., and Mulier, F., "*Learning from data : Concepts, Theory, and Methods*," John Wiley & Sons, New York, 1998.

[16] Burges, C.J.C., "A Tutorial on Support Vector Machines for Pattern Recognition," *Data Mining and Knowledge Discovery*, Vol.2, No.2, pp.121-167, 1998.

[17] Schölkopf, B., "*Advances in Kernel Methods Support Vector Learning*," MIT Press, Cambridge, Mass., 1999.

[18] Vapnik, V.N., "*The Nature of Statistical Learning Theory*," Springer, New York, 2000.

[19] Kecman, V., "*Learning and Soft Computing: Support Vector Machines, Neural Networks, and Fuzzy Logic Models*," MIT Press, Cambridge, Mass., 2001.

[20] Schölkopf, B., and Smola, A.J., "*Learning with Kernels : Support Vector Machines, Regularization, Optimization, and Beyond*," MIT Press, Cambridge, Mass., 2002.

[21] Basudhar, A., and Missoum, S., "Adaptive Explicit Decision Functions for Probabilistic Design and Optimization using Support Vector Machines," *Computers & Structures*, Vol.86, No.19-20, pp.1904-1917, 2008.

[22] Basudhar, A., and Missoum, S., "An Improved Adaptive Sampling Scheme for the Construction of Explicit Boundaries," *Structural and Multidisciplinary Optimization*, Vol.42, No.4, pp.517-529, 2010.

[23] Canu, S., Grandvalet, Y., and Guigue, V., "SVM and Kernel Methods Matlab Toolbox," http://*asi.insa-rouen.fr/enseignants/~arakotom/toolbox/index.html*, 2005.

[24] Viana, F.A.C., "SURROGATES Toolbox User's Guide," http://*sites.google.com/site/fchegury/surrogatestoolbox*, 2010.

[25] Martin, J. D., "Computational Improvements to Estimating Kriging Metamodel Parameters," *Journal of Mechanical Design*, Vol.131, No.8, 2009.

[26] Lewis, R. M., and Torczon, V., "Pattern Search Algorithms for Bound Constrained Minimization," *SIAM Journal on Optimization*, Vol.9, No.4, pp.1082-1099, 1999.

[27] Saka, Y., Gunzburger, M., and Burkardt, J., "Latinized, Improved LHS, and CVT Point Sets in Hypercubes," *International Journal of Numerical Analysis and Modeling*, Vol.4, No.3-4, pp.729-743, 2007.

[28] Coleman, T.F., and Li, Y., "An Interior, Trust Region Approach for Nonlinear Minimization Subject to Bounds," *SIAM Journal on Optimization*, Vol.6, pp.418-445, 1996.

[29] Coleman, T.F., and Li, Y., "On the Convergence of Reflective Newton Methods for Large-Scale Nonlinear

Minimization Subject to Bounds," *Mathematical Programming*, Vol.67, No.2, pp.189-224, 1994.

[30] Powell, M.J.D., "A Fast Algorithm for Nonlinearly Constrained Optimization Calculations," *Numerical Analysis*, ed. G.A. Watson, *Lecture Notes in Mathematics*, Springer Verlag, Vol.630, 1978.

[31] Powell, M.J.D., "The Convergence of Variable Metric Methods for Nonlinearly Constrained Optimization Calculations," *Nonlinear Programming* 3 (Mangasarian, O.L., Meyer, R.R., and Robinson, S.M., eds.), Academic Press, 1978.

[32] Byrd, R.H., Gilbert, J.C., and Nocedal, J., "A Trust Region Method Based on Interior Point Techniques for Nonlinear Programming," *Mathematical Programming*, Vol.89, No.1, pp.149-185, 2000.

[33] Waltz, R.A., Morales, J.L., Nocedal, J., and Orban, D., "An interior algorithm for nonlinear optimization that combines line search and trust region steps," *Mathematical Programming*, Vol.107, No.3, pp.391-408, 2006.